

Assignment-2

MongoDB Schema Design Solutions

1. E-Commerce Store - Products & Orders

users Collection:

```
{
  $jsonSchema: {
    bsonType: "object",
    required: ["name", "email", "password"],
    properties: {
      name: { bsonType: "string" },
      email: {
        bsonType: "string",
        pattern: "^.+@.+\\.+\\.+$",
        description: "Must be a valid email"
      },
      password: {
        bsonType: "string",
        minLength: 6
      }
    }
  }
}
```

products Collection:

```
{
  $jsonSchema: {
    bsonType: "object",
    required: ["title", "description", "price", "category", "stock"],
    properties: {
      title: { bsonType: "string" },
      description: { bsonType: "string" },
      price: { bsonType: "double", minimum: 0 },
      category: {
        bsonType: "string",
        enum: ["Electronics", "Clothing", "Books", "Home", "Other"]
      },
      stock: { bsonType: "int", minimum: 0 }
    }
  }
}
```

orders Collection:

```
{
  $jsonSchema: {
    bsonType: "object",
    required: ["userId", "products", "totalAmount", "orderDate"],
    properties: {
      userId: { bsonType: "objectId" },
      products: {
```

```

        bsonType: "array",
        items: {
            bsonType: "object",
            required: ["productId", "quantity"],
            properties: {
                productId: { bsonType: "objectId" },
                quantity: { bsonType: "int", minimum: 1 }
            }
        },
        totalAmount: { bsonType: "double", minimum: 0 },
        orderDate: { bsonType: "date" }
    }
}

reviews Collection:
{
    $jsonSchema: {
        bsonType: "object",
        required: ["userId", "productId", "rating", "comment"],
        properties: {
            userId: { bsonType: "objectId" },
            productId: { bsonType: "objectId" },
            rating: { bsonType: "int", minimum: 1, maximum: 5 },
            comment: { bsonType: "string" }
        }
    }
}

# Email uniqueness (apply separately):
db.users.createIndex({ email: 1 }, { unique: true })

```

2. Online Course Platform - Instructors & Students

```

users Collection:
{
    $jsonSchema: {
        bsonType: "object",
        required: ["name", "email", "role"],
        properties: {
            name: { bsonType: "string" },
            email: { bsonType: "string", pattern: "^.+@.+\\.+$" },
            role: { bsonType: "string", enum: ["student", "instructor"] }
        }
    }
}

courses Collection:
{
    $jsonSchema: {
        bsonType: "object",
        required: ["title", "instructorId", "category", "price", "createdAt"],
    }
}

```

```

    properties: {
      title: { bsonType: "string" },
      instructorId: { bsonType: "objectId" },
      category: { bsonType: "string" },
      price: { bsonType: "double", minimum: 0 },
      createdAt: { bsonType: "date" }
    }
  }
}

```

```

lessons (embedded):
lessons: {
  bsonType: "array",
  items: {
    bsonType: "object",
    required: ["title", "videoURL", "duration"],
    properties: {
      title: { bsonType: "string" },
      videoURL: { bsonType: "string" },
      duration: { bsonType: "int", minimum: 1 }
    }
  }
}

```

```

enrollments Collection:
{
  $jsonSchema: {
    bsonType: "object",
    required: ["userId", "courseId", "enrolledAt"],
    properties: {
      userId: { bsonType: "objectId" },
      courseId: { bsonType: "objectId" },
      enrolledAt: { bsonType: "date" }
    }
  }
}

```

3. Event Booking System - Organizers & Attendees

```

users Collection:
{
  $jsonSchema: {
    bsonType: "object",
    required: ["name", "email", "role"],
    properties: {
      name: { bsonType: "string" },
      email: { bsonType: "string", pattern: "^.+@.+\\.+$" },
      role: { bsonType: "string", enum: ["organizer", "attendee"] }
    }
  }
}

```

events Collection:

```

{
  $jsonSchema: {
    bsonType: "object",
    required: ["title", "organizerId", "location", "startTime", "endTime", "capacity"],
    properties: {
      title: { bsonType: "string" },
      organizerId: { bsonType: "objectId" },
      location: { bsonType: "string" },
      startTime: { bsonType: "date" },
      endTime: { bsonType: "date" },
      capacity: { bsonType: "int", minimum: 1 }
    }
  }
}

```

bookings Collection:

```

{
  $jsonSchema: {
    bsonType: "object",
    required: ["eventId", "attendeeId", "bookingDate"],
    properties: {
      eventId: { bsonType: "objectId" },
      attendeeId: { bsonType: "objectId" },
      bookingDate: { bsonType: "date" }
    }
  }
}

```

4. Blogging Platform - Authors & Articles

authors Collection:

```

{
  $jsonSchema: {
    bsonType: "object",
    required: ["name", "email", "bio"],
    properties: {
      name: { bsonType: "string" },
      email: { bsonType: "string", pattern: "^.+@.+\\.+\\.+$" },
      bio: { bsonType: "string" }
    }
  }
}

```

articles Collection:

```

{
  $jsonSchema: {
    bsonType: "object",
    required: ["title", "content", "authorId", "tags", "published", "createdAt"],
    properties: {
      title: { bsonType: "string" },
      content: { bsonType: "string" },
      authorId: { bsonType: "objectId" },
      tags: { bsonType: "array", items: { bsonType: "string" } },

```

```

        published: { bsonType: "bool" },
        createdAt: { bsonType: "date" }
    }
}

```

comments Collection:

```

{
  $jsonSchema: {
    bsonType: "object",
    required: ["articleId", "userName", "commentText", "postedAt"],
    properties: {
      articleId: { bsonType: "objectId" },
      userName: { bsonType: "string" },
      commentText: { bsonType: "string" },
      postedAt: { bsonType: "date" }
    }
  }
}

```

5. Subscription App - Users & Plans

users Collection:

```

{
  $jsonSchema: {
    bsonType: "object",
    required: ["email", "name", "signupDate"],
    properties: {
      name: { bsonType: "string" },
      email: { bsonType: "string", pattern: "^.+@.+\\.+\\.+$" },
      signupDate: { bsonType: "date" }
    }
  }
}

```

plans Collection:

```

{
  $jsonSchema: {
    bsonType: "object",
    required: ["name", "price", "features", "billingCycle"],
    properties: {
      name: { bsonType: "string" },
      price: { bsonType: "double", minimum: 0 },
      features: { bsonType: "array", items: { bsonType: "string" } },
      billingCycle: { bsonType: "string", enum: ["monthly", "yearly"] }
    }
  }
}

```

subscriptions Collection:

```

{
  $jsonSchema: {
    bsonType: "object",

```

```
required: ["userId", "planId", "startDate", "isActive"],
properties: {
  userId: { bsonType: "objectId" },
  planId: { bsonType: "objectId" },
  startDate: { bsonType: "date" },
  isActive: { bsonType: "bool" }
}
}
```